

## 1. INTRODUCTION

Scratch is a new programming language that makes it easy to create interactive stories, games, and animations – and share your creations with others on the web.

This Reference Guide provides an overview of the Scratch software. If you are just getting started with Scratch, we encourage you to try the Getting Started Guide first (<http://scratch.mit.edu/files/ScratchGettingStarted.pdf>). Then, if you want more detailed information, come back to the Reference Guide.

The Scratch website has many other resources to help you learn Scratch: Video Tutorials, Scratch Cards, and Frequently Asked Questions (FAQs). See <http://scratch.mit.edu/howto>

For the latest version of this Reference Guide, see <http://scratch.mit.edu/files/ScratchReferenceGuide.pdf>

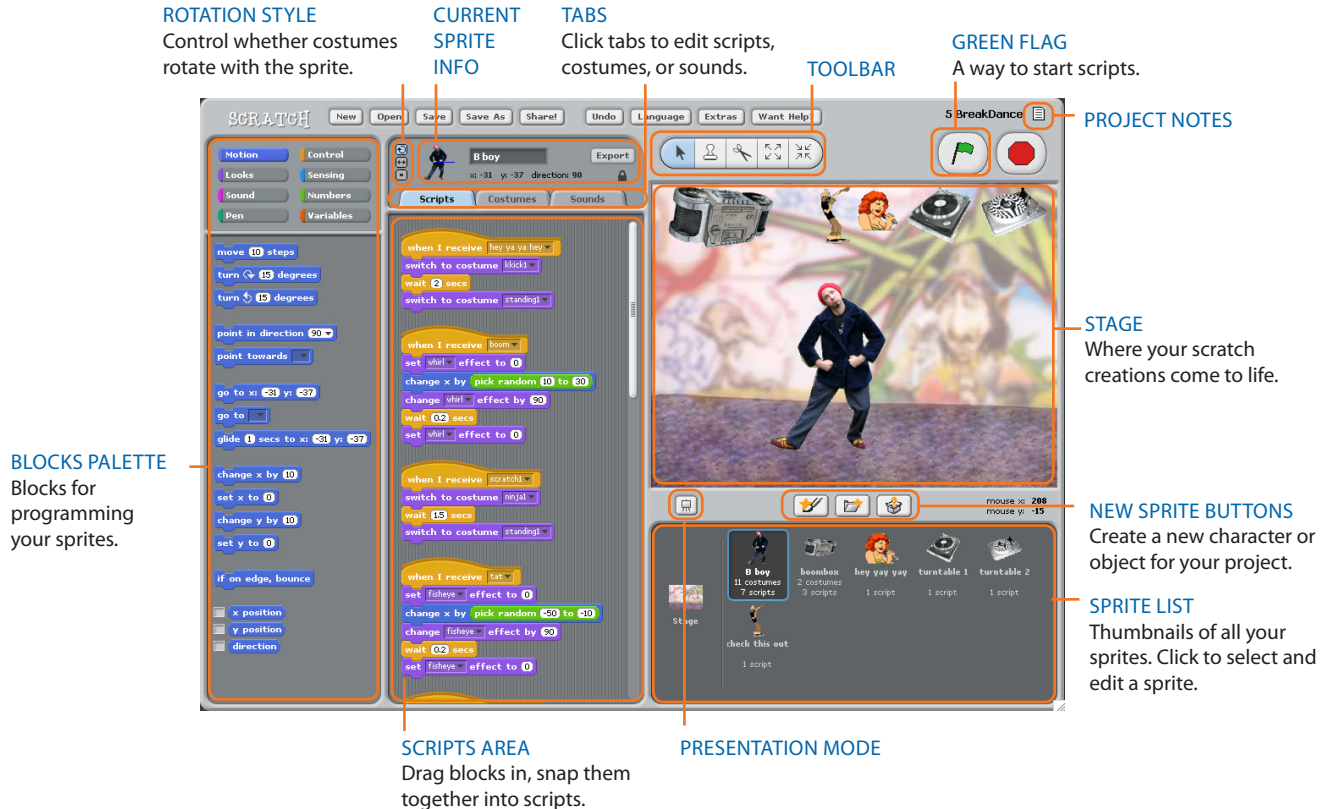
## BASIC INGREDIENTS OF A SCRATCH PROJECT

Scratch projects are made up of objects called **sprites**. You can change how a sprite looks by giving it a different **costume**. You can make a sprite look like a person or a train or a butterfly or anything else. You can use any image as a costume: you can draw an image in the Paint Editor, import an image from your hard disk, or drag in an image from a website.

You can give instructions to a sprite, telling it to move or play music or react to other sprites. To tell a sprite what to do, you snap together graphic **blocks** into stacks, called **scripts**. When you double-click on a script, Scratch runs the blocks from the top of the script to the bottom.

*Scratch is developed by the Lifelong Kindergarten Group at the MIT Media Lab, with financial support from the National Science Foundation, Intel Foundation, and the MIT Media Lab research consortia.*

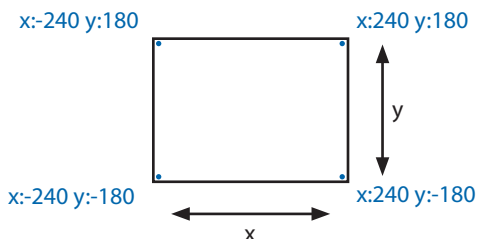
## 2. SCRATCH INTERFACE



### STAGE

The **Stage** is where you see your stories, games, and animations come to life. Sprites move and interact with one another on the Stage.

The Stage is 480 units wide and 360 units tall. It is divided into an x-y grid. The middle of the Stage has an x-coordinate of 0 and a y-coordinate of 0.



mouse x: 80  
mouse y: -25

To find out x-y positions on the Stage, move the mouse around and look at the **mouse x-y display** just below the Stage, on the right.



Click the **Presentation Mode** button to see projects at full-screen size. To exit Presentation Mode, press the Esc key.

## NEW SPRITE BUTTONS

When you start a new Scratch project, it begins with a single cat sprite. To create new sprites, click on these buttons:



Paint your own costume for a new sprite using the Paint Editor.



Select a costume for a new sprite – or import an entire sprite.



Get a surprise sprite.

If you want to delete a sprite, select the scissors from the Toolbar and click on the sprite. Or right-click on the sprite and select *delete* from the pop-up menu.

To make a sprite that looks like part of the Stage background, right-click the Stage and select *grab screen region for new sprite*.

## SPRITE LIST

The **Sprite List** displays thumbnails of all of the sprites in the project. For each sprite, it shows the sprite's name, how many scripts it has, and how many costumes it has.



To see and edit a sprite's scripts, costumes, and sounds, click on the sprite's thumbnail in the Sprite List – or double-click on the sprite itself on the Stage. (The selected sprite is highlighted and outlined in blue in the Sprite List.) To show, export, duplicate, or delete a sprite, right-click on the sprite's thumbnail in the Sprite List. To show a sprite that is off the stage or hidden, Shift+click on the sprite's thumbnail in the Sprite List - this will bring the sprite to the middle of the Stage and show it.

You can rearrange the sprites in the Sprite List by dragging the thumbnails.

Just as a sprite can change its appearance by switching costumes, the Stage can change its appearance by switching **backgrounds**. To see and edit the scripts, backgrounds, and sounds associated with the Stage, click on the Stage icon at the left of the Sprite List.

## BLOCKS PALETTE and SCRIPTS AREA

To program a sprite, drag out blocks from the **Blocks Palette** to the **Scripts Area**. To run a block, double-click on it.

Create scripts (programs) by snapping blocks together into stacks. Double-click anywhere on the stack to run the whole script, from top to bottom. To find out what a block does, right-click on it, then select *help* from the pop-up menu.

When you drag a block around the Scripts Area, a white highlight indicates where you can drop the block and form a valid connection with another block. You can insert blocks in the middle of a stack or at the end.

To move a stack, pick it up from the top block. If you drag out a block from the middle of a stack, all of the blocks beneath it will come along with it. To copy a stack of blocks from one sprite to another, drag the stack to the thumbnail of the other sprite in the Sprite List.

Some blocks have white editable text fields inside, such as `move 10 steps`. To change the value, click inside the white area and edit the number. You can also drop rounded blocks, like `x position`, inside these areas.

Some blocks have pull-down menus, such as `set instrument to 1`. Just click on the ▼ to see the menu, then click again to make a selection.

To clean up the Scripts Area, right-click and select *clean up* from the menu. To export a screenshot of the Scripts Area, right-click and select *save picture of scripts*.

## COSTUMES

Click the **Costumes** tab to see and edit the sprite's costumes.



This Sprite has two costumes. The sprite's current costume (girl1-walking) is highlighted. To switch to a different costume, simply click on the thumbnail of the costume you want.

There are three ways to create new costumes:

- Click **Paint** to paint a new costume in the Paint Editor.
- Click **Import** to import an image file from your hard disk.
- Drag in one or more images from the web or your desktop.

Scratch can recognize many different image formats: JPG, BMP, PNG, and GIF (including animated GIF).

Each costume has a costume number (displayed to its left). You can rearrange the order of the costumes by dragging the thumbnails. The costume numbers will update if you change their order.

Right-click on a costume thumbnail to convert the costume into a new sprite, or to export a copy of the costume as a separate file.

## SOUNDS

Click the **Sounds** tab to see the sprite's sounds.



You can record new sounds and import sound files. Scratch can read MP3 files and uncompressed WAV, AIF, and AU files (encoded with 8-bits or 16-bits per sample, but not 24-bits per sample).

## CURRENT SPRITE INFO



**Current Sprite Info** shows a sprite's name, x-y position, direction, and lock state.

You can type in a new name for the sprite.

The sprite's direction indicates which direction the sprite will move when it runs a move block (0=up, 90=right, 180=down, -90=left). The blue line on the thumbnail shows the sprite's direction. You can drag this line to change the sprite's direction. Double-click on the sprite to set the direction back to its original state (direction=90).

Click the **Lock** to change the sprite's lock state. An unlocked sprite can be dragged in presentation mode and the web player.

Click **Export** to save the sprite as a separate file, allowing it to be imported into another project.

## ROTATION STYLE

Click the **Rotation Style** buttons to control how the costume appears as the sprite changes its direction.

- ☒ Rotate: The costume rotates as the sprite changes direction.
- ☐ Left-right flip: The costume faces either left or right.
- ☐ No-rotate: The costume never rotates (even as the sprite changes direction).

## TOOLBAR



Click on the **Toolbar** to select a tool, then click on other objects to perform an action.

- Arrow:** Normal mode. Pick up and move sprites and blocks.
- Duplicate:** Duplicate sprites, costumes, sounds, blocks, and scripts. (Shift+click for multiple.)
- Delete:** Delete sprites, costumes, sounds, blocks, and scripts. (Shift+click for multiple.)
- Grow:** Make sprites bigger. (Shift+click for larger size steps.)
- Shrink:** Make sprites smaller. (Shift+click for larger size steps.)

## MENU



**New**, **Open**, **Save**, and **Save As** do what you would expect them to do.

**Share!** uploads your project to the Scratch website (<http://scratch.mit.edu>).

**Undo** allows you to retrieve the last block, script, or sprite you deleted (but does not allow you to undo most other actions).

**Language** allows you to select the language of the user interface. While complete translations are provided for some languages, only the scripts and command blocks are translated for others (improved translations will be released on the Scratch website as they become available). Anyone can add or edit language translations of Scratch. To add or change a translation, simply edit the files in the *locale* folder within the top level of the *Scratch* folder. (Please refer to the *Scratch.pot* file for further instructions.) Translations are currently limited to the Latin character set.

**Extras** gives a pop-up menu with special features:

**Import Project:** Bring all of the sprites and backgrounds from another project into this project. This feature is useful for combining sprites from multiple projects.

**Start Single Stepping:** The Scratch program runs one step at a time, highlighting each block as it runs. This feature can be useful for finding bugs in programs, and for helping new programmers understand the flow of a program.


**Compress Sounds:** Compress sounds used in the project, to reduce the overall file size of the project.

**Compress Images:** Compress images used in the project, to reduce the overall file size of the project.

**Want Help?** brings up a page with links to reference materials, tutorials, and frequently asked questions. To get help on an individual block, right-click on the block and select *help* from the pop-up menu.

## GREEN FLAG

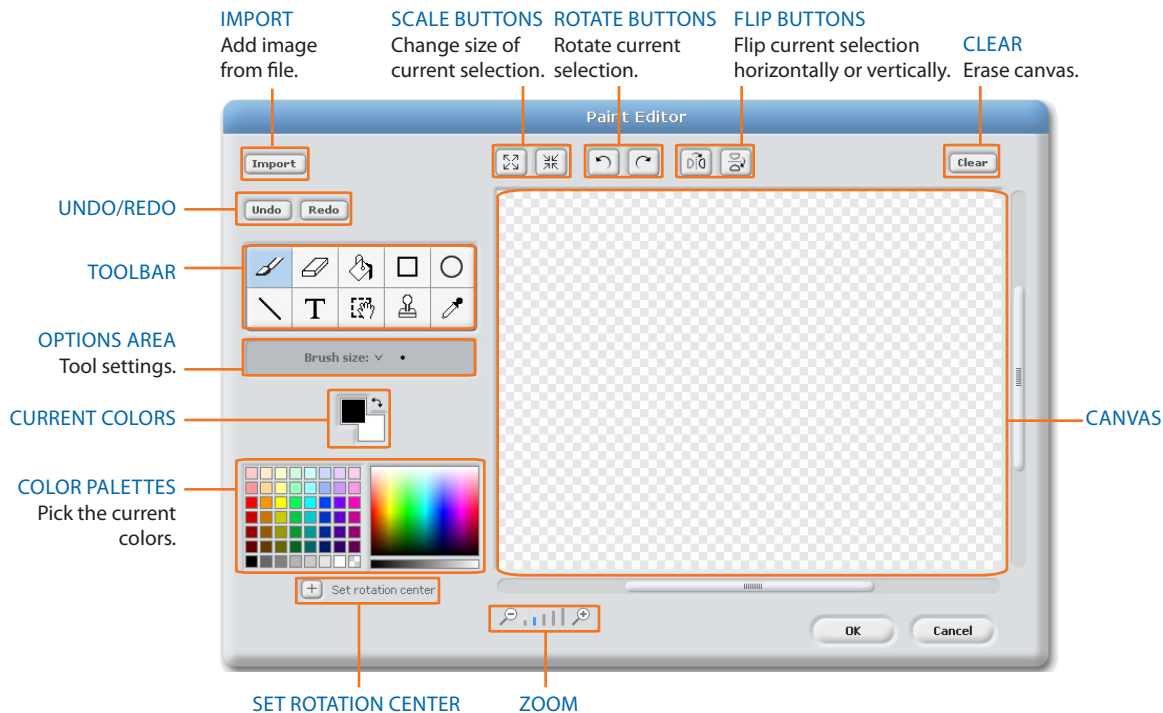
The **Green Flag** provides a convenient way to start many scripts at the same time. Click the Green Flag to start all scripts with  at the top.

In Presentation Mode, the Green Flag appears as a tiny icon  at the top-right corner of the screen. Pressing the Enter key has the same effect as clicking the Green Flag.



## PAINT EDITOR

You can use the **Paint Editor** to create or edit costumes and backgrounds.



The Paint Editor **Toolbar** has the following tools:

**Paintbrush**: Paint freehand using the current foreground color. When you click on this tool, the Options Area shows the brush size. Click ☐ to pick a different brush size.

**Eraser**: Erase with freehand strokes. The areas that you erase become transparent. When you click on this tool, the Options Area shows the eraser size. Click ☐ to pick a different eraser size.

**Fill**: Fill connected areas with a solid color or gradient. When you click this tool, the Options Area shows the fill style (*solid color*, *horizontal gradient*, *vertical gradient*, or *radial gradient*). Gradients blend from the selected foreground color to the selected background color.

**Rectangle**: Draw a filled or outlined rectangle (Shift+drag for a square) using the current foreground color. When you click this tool, the Options Area shows the fill style (*solid* or *outlined*).

**Ellipse**: Draw a filled or outlined ellipse (Shift+drag for a circle) using the current foreground color. When you click this tool, the Options Area shows the fill style (*solid* or *outlined*).



**Line:** Draw a straight line (Shift+drag for a horizontal or vertical line) using the current foreground color. When you click on this tool, the Options Area shows the brush size.

Click  to pick a different brush size.

**Text:** Add text to the drawing. When you click on this tool, the Options Area allows you to change the font face and font size. Each costume can only have one block of text.

**Selection:** Select a rectangular region, then move it to a new location (press delete to remove the selection, Shift+delete or Shift+backspace to crop).

**Stamp:** Select a rectangular region, then copy it to new locations (Shift+click+drag for repeated stamping).

**Eyedropper:** Use the tip of the eyedropper to choose the foreground color (click on the Canvas and drag the eyedropper to pick a color from outside the Canvas).

The **Current Colors** (foreground and background) are shown below the Options Area. You can click the color swap arrow to exchange the foreground and background colors. Click in one of the **Color Palettes** to pick a new foreground color (Shift+click to select a background color).

Click the **Set Rotation Center** button to select the location within the drawing that will be used as the center of rotation when the Costume is rotated on the Stage.

Click the **Zoom** buttons (in or out) to increase or decrease the view magnification of the Canvas. When the zoom is greater than 100%, scroll bars are used to pan around the Canvas. Zoom does not change the size of the image.

Click on the **Import** button to open an image from file and add it to the Canvas.

Click the **Clear** button to remove all content from the Canvas.

To change the size of the Canvas content, or just the current selection, click on the **Scale** buttons (grow or shrink). You can Shift+click on the buttons to enter a precise value. Shrinking decreases the size and resolution of the image.

To rotate the Canvas content, or just the current selection, click on the **Rotate** buttons (counter-clockwise or clockwise). You can Shift+click on the buttons to enter a precise value.





To flip the Canvas content, or just the current selection, click on the **Flip** buttons (horizontal or vertical).


If you make a mistake, you can click the **Undo** button repeatedly to undo the last several actions. If you change your mind, you can click the **Redo** button to restore the undone actions.




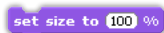


## 3. SCRATCH BLOCKS


### TYPES OF BLOCKS




There are three main types of blocks in the **Blocks Palette**:

**Stack Blocks:** These blocks have bumps on the bottom and/or notches on the top, such as . You can snap these blocks together into stacks. Some stack blocks have an input area inside them, where you can type a number (such as **10** in the  block) or choose an item from a pull-down menu (such as *pop* in the  block). Some stack blocks, such as , have a C-shaped “mouth” where you can insert other stack blocks.

**Hats:** These blocks have rounded tops, such as . These blocks are placed at the tops of stacks. They wait for an event to happen, such as a key being pressed, then run the blocks underneath them.

**Reporters:** These blocks, such as  and , are designed to fit in the input area of other blocks. Reporters come in two shapes, and fit only into “holes” of the same shape. Reporters with rounded ends (such as ) report *numbers* and fit inside blocks with rounded holes (such as ). Reporters with pointed ends (such as ) report *boolean values* (*true* or *false*) and fit inside blocks with pointed holes (such as ).

Some reporter blocks have a check box next to them, such as . If you click in the check box, a **monitor** appears on the stage, displaying the current value of the reporter. As the value of the reporter changes, the monitor updates automatically. A monitor can display the value of the reporter in several different formats:

-  a small readout with the name of the reporter
-  a large readout without any name
-  a slider that allows you to manipulate the value of the reporter (available only for variables)

Double-click or right-click on the monitor to change from one format to another.









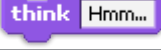
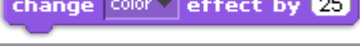

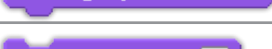
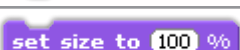



The slider format is available only for user-created variables. Right-click on the monitor in slider format to adjust its minimum and maximum values.



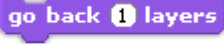
## BLOCK DESCRIPTIONS

The Scratch blocks are organized into eight color-coded categories: motion, looks, sound, pen, control, sensing, numbers, and variables.






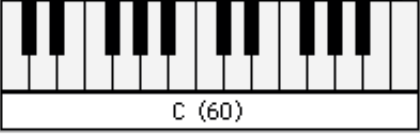







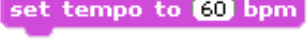


Motion	
	Moves sprite forward or backward.
	Rotates sprite clockwise.
	Rotates sprite counterclockwise.
	Points sprite in the specified direction. (0=up, 90=right, 180=down, -90=left)
	Points sprite toward mouse-pointer or another sprite.
	Moves sprite to specified x and y position on Stage.
	Moves sprite to the location of the mouse-pointer or another sprite.
	Moves sprite smoothly to a specified position over specified length of time.
	Changes sprite's x-position by specified amount.
	Sets sprite's x-position to specified value.
	Changes sprite's y-position by specified amount.
	Sets sprite's y-position to specified value.
	Turns sprite in opposite direction when sprite touches edge of Stage.
	Reports sprite's x-position. (Ranges from -240 to 240)
	Reports sprite's y-position. (Ranges from -180 to 180)
	Reports sprite's direction. (0=up, 90=right, 180=down, -90=left)

## Looks












	Changes sprite's appearance by switching to different costume.
	Changes sprite's costume to next costume in the costume list. (If at end of the costume list, jumps back to first costume.)
	Reports sprite's current costume number.
	Changes Stage's appearance by switching to a different background.
	Changes Stage's background to next background in the background list.
	Reports Stage's current background number.
	Displays sprite's speech bubble for specified amount of time.
	Displays sprite's speech bubble. (You can remove speech bubble by running this block without any text.)
	Displays sprite's thought bubble for specified amount of time.
	Displays sprite's thought bubble.
	Changes a visual effect on a sprite by specified amount. (Use pull-down menu to choose effect.)
	Sets a visual effect to a given number. (Most visual effects range from 0 to 100.)
	Clears all graphic effects for a sprite.
	Changes sprite's size by specified amount.
	Sets sprite's size to specified % of original size.
	Reports sprite's size, as % of original size.
	Makes sprite appear on the Stage.

	Makes sprite disappear from the Stage. (When sprite is hidden, other sprites cannot detect it with touching? block.)
	Moves sprite in front of all other sprites.
	Moves sprite back a specified number of layers, so that it can be hidden behind other sprites.







## Sound











	Starts playing a sound, selected from pull-down menu, and immediately goes on to the next block even as sound is still playing.
	Plays a sound and waits until the sound is finished playing before continuing with next block.
	Stops playing all sounds.
	Plays a drum sound, selected from pull-down menu, for specified number of beats.
 	Plays a musical note (higher numbers for higher pitches) for specified number of beats.
	Rests (plays nothing) for specified number of beats.
	Sets the type of instrument that the sprite uses for <b>play note</b> blocks. (Each sprite has its own instrument.)
	Changes sprite's sound volume by specified amount.
	Sets sprite's sound volume to specified value.
 	Reports sprite's sound volume.
	Changes sprite's tempo by specified amount.
	Sets sprite's tempo to specified value in beats per minute.
 	Reports sprite's tempo in beats per minute.

## Pen



	Clears all pen marks and stamps from the Stage.
	Puts down sprite's pen, so it will draw as it moves.
	Pulls up sprite's pen, so it won't draw as it moves.
	Sets pen's color, based on choice from color picker.
	Changes pen's color by specified amount.
	Sets pen's color to specified value. (pen-color=0 at red end of rainbow, pen-color=100 at blue end of rainbow)
	Changes pen's shade by specified amount.
	Sets pen's shade to specified amount. (pen-shade=0 is very dark, pen-shade=100 is very light)
	Changes pen's thickness.
	Sets pen's thickness.
	Stamps sprite's image onto the Stage.

## Control















	Runs script below when green flag is clicked.
	Runs script below when specified key is pressed.
	Runs script below when sprite is clicked.
	Waits specified number of seconds, then continues with next block.
	Runs the blocks inside over and over.
	Runs the blocks inside a specified number of times.

	Sends a message to all sprites, triggering them to do something, and waits until they all finish before continuing with next block.
	Sends a message to all sprites, then continues with the next block without waiting for the triggered scripts.
	Runs script below when it receives specified broadcast message.
	Continually checks whether condition is true; whenever it is, runs the blocks inside.
	If condition is true, runs the blocks inside.
	If condition is true, runs the blocks inside the <b>if</b> portion; if not, runs the blocks inside the <b>else</b> portion.
	Waits until condition is true, then runs the blocks below.
	Checks to see if condition is true; if so, runs blocks inside and checks condition again. If condition is not true, goes on to the blocks that follow.
	Stops the script.
	Stops all scripts in all sprites.








Sensing	
	Reports the x-position of mouse-pointer.
	Reports the y-position of mouse-pointer.
	Reports true if mouse button is pressed.
	Reports true if specified key is pressed.
	Reports true if sprite is touching specified sprite, edge, or mouse-pointer. (Selected from pull-down menu.)
	Reports true if sprite is touching specified color. (Click on color patch, then use eyedropper to select color.)
	Reports true if first color (within sprite) is touching second color (in background or another sprite). (Click on color patch, then use eyedropper to select color.)
	Reports distance from the specified sprite or mouse-pointer.
	Sets the timer to zero.
	Reports the value of the timer in seconds. (The timer is always running.)
	Reports a property or variable of another sprite.
	Reports the volume (from 1 to 100) of sounds detected by the computer microphone.
	Reports true if computer microphone detects a sound volume greater than 30 (on scale of 1 to 100).
	Reports the value of specified sensor. To use this block, you need to have a Scratch Board connected to your computer. See <a href="http://scratch.mit.edu/scratchboard">http://scratch.mit.edu/scratchboard</a>
	Reports true if specified sensor is pressed. To use this block, you need to have a Scratch Board connected to your computer. See <a href="http://scratch.mit.edu/scratchboard">http://scratch.mit.edu/scratchboard</a>

## Numbers

	Adds two numbers.
	Subtracts second number from first number.
	Multiplies two numbers.
	Divides first number by second number.
	Picks a random integer within the specified range.
	Reports true if first number is less than second.
	Reports true if two numbers are equal.
	Reports true if first number is greater than second.
	Reports true if both conditions are true.
	Reports true if either condition is true.
	Reports true if condition is false; reports false if condition is true.
	Reports result of selected function (abs, sqrt, sin, cos, tan, asin, acos, atan, ln, log, e^, 10^ ) applied to specified number.
	Reports remainder from division of first number by second number.
	Reports closest integer to a number.

## Variables

	Allows you to create and name a new variable. When you create a variable, three blocks are created automatically (see below). You can choose whether the variable is for all sprites (global) or just for one sprite (local).
	Deletes all three blocks associated with the variable.
	Changes variable by specified amount.
	Sets variable to specified number.
	Reports value of variable.